ERDC MSRC PET Technical Report No. 01-08

Visualization for EQM

by

M. Polly Baker
Alan M. Shih

27 April 2001

Work funded by the Department of Defense
High Performance Computing Modernization Program
U.S. Army Engineer Research and Development Center
Major Shared Resource Center through

Programming Environment and Training

Supported by Contract Number: DAHC94-96-C0002
Computer Sciences Corporation

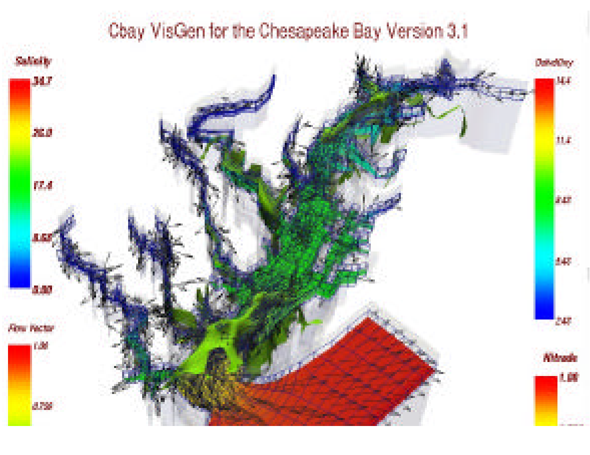
Visualization for EQM

Alan M. Shih
Research Scientist, NCSA, University of Illinois

M. Pauline Baker
Division Director, Visualization, NCSA, University of Illinois

1. Introduction

The numerical simulation of water quality in the Chesapeake Bay is an important on-going project at the U.S. Army Engineer Research and Development Center (ERDC). ERDC researchers, led by Dr. Carl F. Cerco, have developed and refined the numerical simulation models (1) over several years, and are now able to accomplish long-term scenario runs, that is, scenarios that span more than 10 years of water quality variation. The simulation output from these runs is large and complex. Dr. Cerco's team has been working with the visualization team at the National Center for Supercomputing Applications (NCSA) at the University of Illinois through the ERDC MSRC PET program to develop a customized visualization tool for these datasets. Over the course of the collaboration, a visualization tool called CbayVisGen (Version 3.1, see Figure 1) was developed to support the visualization needs of Dr. Cerco's team.



In a focused effort carried out during Year 5, this team worked to support Dr. Cerco's ERDC EQM team in using the current CbayVisGen tool. The NCSA PET team also enhanced the tool to include the visualization of observational data collected over many years in the Chesapeake Bay. Further, the NCSA team explored the possibilities for addressing issues related to the visualization of very large data sets. Many traditional visualization tools operate as a single executable, and assume that the user will download the data to their local workstation. In scenario runs, however, the data is simply too large to download to the local workstation. In newer work, the NCSA team is experimenting with building interactive tools that work in a distributed fashion. In this focused effort, we apply these techniques to the visualization of Chesapeake Bay simulation output and data.

2. Technical Approaches and Accomplishments

CBayVisGen depends on VTK (the Visualization ToolKit) for its visualization and rendering algorithms (2,3). VTK is an open-source, freely available software system for computer graphics, image processing, and visualization. It supports hundreds of algorithms in the visualization and image processing fields, and is widely used in the visualization community.

In CbayVisGen, a message-passing scheme is used where messages pass from the application layer to the core algorithms. As a result of these messages, the core algorithm produces a set of graphical objects that are available to the application for rendering. Since these messages are simple strings, it is very easy to use the resulting tool in a collaborative fashion, where application states are shared between multiple users. In addition, this type of message passing also easily supports rapid prototyping of new data representations, remote visualization, and easy integration with components of the user interface. The design of this tool allows a user to add graphical objects to the screen, such as isosurfaces, vectors, and discrete data points.

The Fast Light Toolkit (FLTK) is used as the graphical user interface (4). The use of FLTK allowed us to build the user interface panels very quickly and it integrated with the VTK application seamlessly. FLTK is also an open-source, freely available software system.

During Year 5, through this focused effort, we provided assistance in the use of the CbayVisGen Version 3.1 tool. We also modified this tool to meet the user team's special visualization needs such as the color map arrangement. The researcher team applied this tool to explore data, and to obtain images for paper publications, presentations, and part of a book.

quantitatively with spheres, with the sphere radius showing the magnitude of the observed scalar at that point. The isosurface and other visualization idioms represent the modeled data.

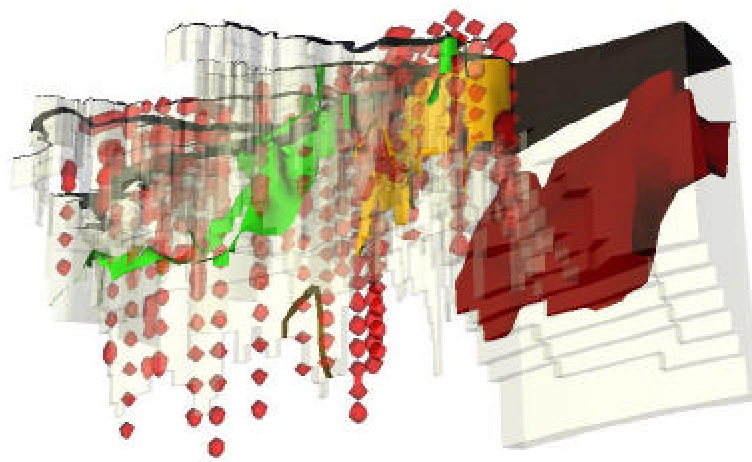
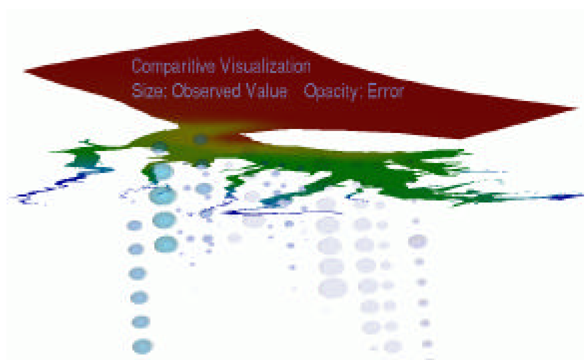


Figure 2. In this comparative visualization, the size and opacity of the spheres are used to show the values of observed data.

Shown in Figure 3 is comparative visualization where the sphere represents the errors between the observed data and the modeled data. The size of a sphere represents the scalar magnitude and the opacity represents the error. Therefore, the less visible spheres represent a smaller error. The error is the difference of scalar values between the observed data and the simulation data in the numerical cell where the observed data point is spatially located.



Figures 4(a)-4(c) demonstrates the sequence of observed data being taken along the time line. During a data collection cruise, scientists travel by boat to each observation point in the bay and take field measurements of water properties. The timestamps vary for each observation. To synchronize the simulation output and observed data in terms of time, the CBayVisGen tool was extended this year to allow the user to specify a time margin for the Julian Date of the simulation output. Observed data that fall outside this time range are cropped.

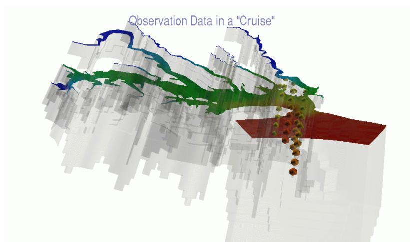


Figure 4(a) Initial Julian Date

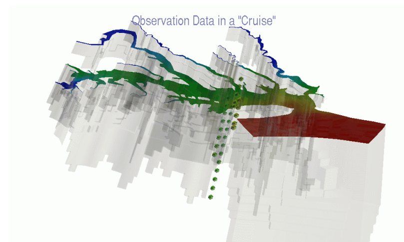


Figure 4(b) Interim Julian Date

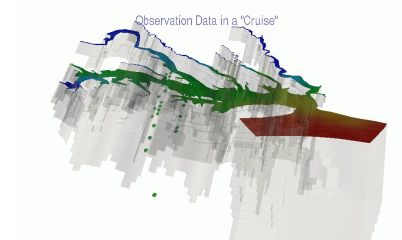
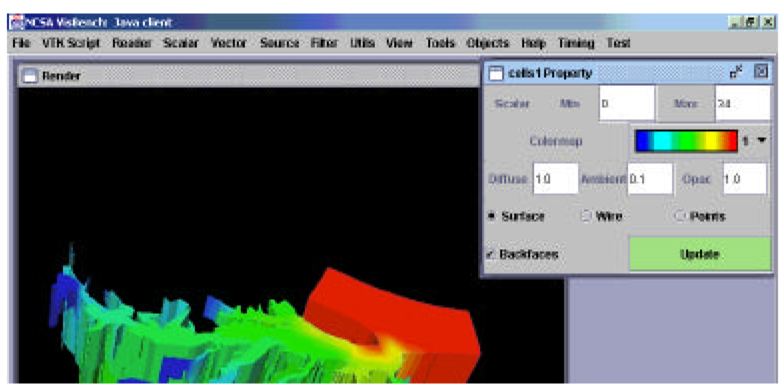
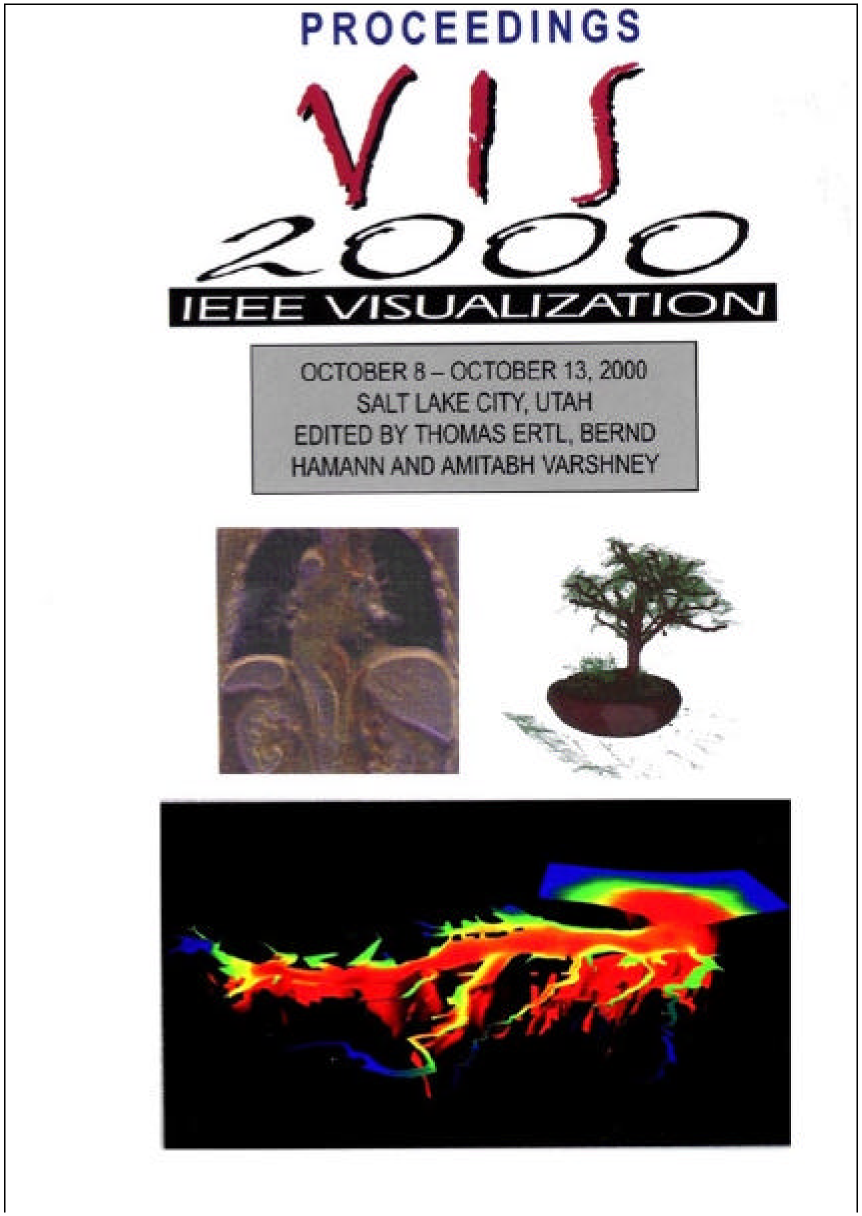


Figure 4(c) As Julian Date Increases

In Year 5, we also explored the possibilities of converting the CBayVisGen tool to work in a client-server manner, such that the data need not be downloaded to the local workstation. To experiment with this idea, we used VisBench (7), an NCSA on-going project sponsored by various agencies, as the testbed. The Chesapeake Bay simulation output was converted to a compatible format. The client program resided on a Pentium II-233 IBM notebook with 96 MB of memory, while the data and server program resided on a Silicon Graphics workstation. The result was very satisfactory and a user can obtain a decent frame rate on the local machine through this client-server arrangement. Figure 5 illustrates the operation of this test and the result on the client screen.



Finally, visualization work from this effort was accepted for publication in the IEEE ***Visualization 2000*** conference. As shown in Figure 6, an image from the paper was used in the front cover of the proceedings.



Acknowledgement

This work was supported in part by a grant of HPC time at ERDC MSRC from the DoD HPC Modernization Program.

References

1. Carl F. Cerco, and Thomas Cole, “Three-Dimensional Eutrophication Model of Chesapeake Bay,” in *Journal of Environment Engineering*, pp. 1006-1025, 1993.
2. Homepage for the Visualization Toolkit (VTK), http://www.kitware.com/vtk.html
3. William J. Schroeder, Kenneth M. Martin, and William E. Lorenson, “The Design and Implementation of an Object-Oriented Toolkit for 3D Graphics and Visualization,” proceedings of the IEEE Visualization 1996, pp. 93-100.
4. Homepage for The Fast Light Toolkit (FLTK), http://www.fltk.org
5. Robert Stein, Alan M. Shih, M. Pauline Baker, Carl F. Cerco, and Mark A. Noel, “Scientific Visualization of Water Quality in the Chesapeake Bay,” proceedings of the IEEE Visualization 2000, 8-13 October 2000, Salt Lake City, Utah, pp. 509-512.
6. Adam Forgang, Bernd Hamman, and Carl F. Cerco, “Visualization of Water Quality Data for the Chesapeake Bay,” in IEEE Visualization 1996, pp. 417-420.
7. Randy Heiland, “VisBench: A Framework for Remote Data Visualization and Analysis,” to appear in the proceedings of the 2001 International Conference on Computational Science, 28-30 May 2001, San Francisco.